AMENDMENTS TO THE CLAIMS

1. – 66. (Canceled).

67. (Currently amended) A method of producing obfuscated object code, the method comprising:
substituting ~~an assignment of~~ a variable in source code with a class template defining a plurality of functions of the variable, each of the plurality of functions indexed by a key value and associated with a series of operations resulting in ~~the~~ <u>an</u> assignment of the variable in a manner obfuscating such assignment<u>, wherein the series of operations reliably returns the variable to an assigned value</u>; and compiling the source code <u>by</u> using a compiler to produce object code, wherein the compiler inserts in the object code the series of operations associated with one of the plurality of functions of the variable identified by a key value provided as a parameter of the class template.

68. (Previously presented) The method according to claim 67, wherein the series of operations comprises a plurality of operations from a group of operations including arithmetic operations and logical operations.

69. – 71. (Canceled)

72. (Previously presented) The method according to claim 67, wherein the source code is written in the C^{++} programming language.

73. – 77. (Canceled)

78. (Currently amended) A method of producing storage media having a secured executable program thereon, the method comprising:
obfuscating object code of a security program by substituting ~~an assignment of~~ a variable in source code of the security program with a class template defining a plurality of functions of the variable, each of the plurality of functions indexed by a key value and associated with a series of operations resulting in ~~the~~ <u>an</u>

assignment of the variable in a manner obfuscating such assignment, and compiling the source code <u>by</u> using a compiler to produce the object code, wherein the compiler inserts in the object code the series of operations associated with one of the plurality of functions of the variable identified by a key value provided as a parameter of the class template<u>, and wherein the series of operations reliably returns the variable to an assigned value</u>;

securing an executable program by associating the executable program with the security program which is arranged to control access to the executable program; and

applying the secured executable program to a storage media.

79. (Previously presented) The method according to claim 78, wherein the executable program and the security program are associated at object code level, the security program being arranged to encrypt the executable program.

80. (Previously presented) The method according to claim 78, further comprising moving blocks of the executable program out of the executable program and relocating the blocks in the security program.

81. (Previously presented) The method according to claim 78, wherein the security program is arranged to require the running of an authentication program.

82. (Previously presented) The method according to claim 78, wherein the source code of the security program involves stored arrays and templates and utilizes pointers to navigate the arrays and templates.

83. (Previously presented) The method according to claim 78, wherein the source code of the security program is written in the C^{++} programming language.

84. (Previously presented) The method according to claim 78, wherein the storage media onto which the secured executable program is applied is an optical disc.

85. (Previously presented) The method according to claim 84, wherein the secured executable program is applied to the optical disc by laser beam encoding.

86. (Previously presented) The method according to claim 78, wherein the storage media onto which the secured executable program is applied is a memory unit associated with at least one computer.

87. (Currently amended) A storage media having an executable program and a security program thereon, wherein the security program controls access to the executable program, the security program is in object code, and the object code of the security program has been obfuscated by substituting an assignment of a variable in source code of the security program with a class template defining a plurality of functions of the variable, each of the plurality of functions indexed by a key value and associated with a series of operations resulting in ~~the~~ an assignment of the variable in a manner obfuscating such assignment, and compiling the source code by using a compiler to produce the object code, wherein the compiler inserted in the object code the series of operations associated with one of the plurality of functions of the variable identified by a key value provided as a parameter of the class template, and wherein the series of operations reliably returns the variable to an assigned value.

88. (Previously presented) The storage media according to claim 87, wherein the series of operations comprises a plurality of operations from a group of operations including arithmetic operations and logical operations.

89. (Canceled)

90. (Previously presented) The storage media according to claim 87, wherein the executable program and the security program are associated at object code level, and wherein the executable program is encrypted on the storage media and the associated security program enables decryption of the executable program.

91. (Previously presented) The storage media according to claim 87, wherein blocks from the executable program have been relocated within the security program.

92. (Previously presented) The storage media according to claim 87, wherein the security program is arranged to require the running of an authentication program.

93. (Previously presented) The storage media according to claim 87, wherein the storage media is an optical disc on which the executable program and the security program are encoded.

94. (Previously presented) The storage media according to claim 93, wherein the optical disc is one of: a CD, a CD-ROM, and a DVD.

95. (Previously presented) The storage media according to claim 87, wherein the storage media is a memory unit associated with at least one computer.

96. (Previously presented) The storage media according to claim 87, wherein the executable program is one or more of:

> a games program;
>
> a video program;
>
> an audio program; and
>
> other software.

97 – 105. (Canceled)